CERTIFIED COPY OF PRIORITY DOCUMENT

EXPRESS MAIL NO.
EV449560063US

REGISTRY OF PATENTS
SINGAPORE

This is to certify that the annexed is a true copy of following application as filed with the Registry.

Date of Filing	:	4 APRIL 2003
Application Number	:	[illegible]1780-3
Applicant(s) / Proprietor(s) of Patent	:	STMICROELECTRONICS ASIA PACIFIC PTE LTD
Title of Invention	:	METHOD AND APPARATUS FOR PERFORMING MODULAR ARITHMETIC

SHARMAINE WU (Ms)
Assistant Registrar
for REGISTRAR OF PATENTS
SINGAPORE

8 APRIL 2004

BEST AVAILABLE COPY

ACTION

PATENTS FORM 1
Patents Act
(Cap. 221)
Patents Rules
Rule 19

INTELLECTUAL PROPERTY OFFICE OF SINGAPORE

REQUEST FOR THE GRANT OF A PATENT UNDER SECTION 25

101101

* denotes mandatory fields

1. YOUR REFERENCE* 1012541/STMicroelectronics/MK/gk

2. TITLE OF INVENTION* METHOD AND APPARATUS FOR PERFORMING MODULAR ARITHMETIC

3. DETAILS OF APPLICANT(S)* (see note 3) Number of applicant(s): 1

(A) Name: STMICROELECTRONICS ASIA PACIFIC PTE LTD
(a private limited company incorporated in the Republic of Singapore)

Address: 28 Ang Mo Kio Industrial Park 2
Singapore 569508

State:

Country: SG

[x] For corporate applicant

State of incorporation:

Country of incorporation: SG

[ ] For individual applicant

State of residency:

Country of residency:

[ ] For others (please specify in the box provided below)

(B) Name:

Address:

State:

Country:

For corporate applicant

State of incorporation

Country of incorporation

For individual applicant

State of residency

Country of residency

For others (please specify in the box provided below)

(C) Name

Address

State

Country

For corporate applicant

State of incorporation

Country of incorporation

For individual applicant

State of residency

Country of residency

For others (please specify in the box provided below)

Further applicants are to be indicated on continuation sheet 1

4. DECLARATION OF PRIORITY (see note 5)

A. Country/country designated

File number

Filing Date DD MM YYYY

B. Country/country designated

File number

Filing Date DD MM YYYY

Further details are to be indicated on continuation sheet 6

5. INVENTOR(S)* (see note 6)

A. The applicant(s) is/are the sole/joint inventor(s) Yes [ ] No [X]

B. A statement on Patents Form 8 is/will be furnished Yes [x] No [ ]

6. CLAIMING AN EARLIER FILING DATE UNDER (see note 7)

[ ] section 20(3) [ ] section 26(6) [ ] section 47(4)

Patent application number [ ]

DD MM YYYY

Filing Date [ ]

Please mark with a cross in the relevant checkbox provided below
(Note: Only one checkbox may be crossed.)

[ ] Proceedings under rule 27(1)(a)

DD MM YYYY

Date on which the earlier application was amended [ ]

[ ] Proceedings under rule 27(1)(b)

7. SECTION 14(4)(C) REQUIREMENTS (see note 8)

Invention has been displayed at an international exhibition. Yes [ ] No [ ]

8. SECTION 114 REQUIREMENTS (see note 9)

The invention relates to and/or used a micro-organism deposited for the purposes of disclosure in accordance with section 114 with a depository authority under the Budapest Treaty.

Yes [ ] No [x]

9. CHECKLIST*

(A) The application consists of the following number of sheets

i.	Request	5	Sheets
ii.	Description	15	Sheets
iii.	Claim(s)	2	Sheets
iv	Drawing(s)	3	Sheets
v.	Abstract (Note: The figure of the drawing, if any, should accompany the abstract)	1	Sheets
	Total number of sheets	26	Sheets

(B) The application as filed is accompanied by.

[ ] Priority document(s) [ ] Translation of priority document(s)

x	Statement of inventorship & right to grant		International exhibition certificate

10. DETAILS OF AGENT (see notes 10, 11 and 12)

Name	
Firm	DONALDSON & BURKINSHAW

11. ADDRESS FOR SERVICE IN SINGAPORE* (see note 10)

Block/Hse No.		Level No.		Unit No./PO Box	3667
Street Name					
Building Name					
Postal Code	905667				

12. NAME, SIGNATURE AND DECLARATION (WHERE APPROPRIATE) OF APPLICANT OR AGENT* (see note 12)
(Note: Please cross the box below where appropriate.)

[X] I, the undersigned, do hereby declare that I have been duly authorised to act as representative, for the purposes of this application, on behalf of the applicant(s) named in paragraph 3 herein.

DONALDSON & BURKINSHAW

Name and Signature

DD MM YYYY

04 04 2003

NOTES:

1. This form when completed, should be brought or sent to the Registry of Patents together with the rest of the application. Please note that the filing fee should be furnished within the period prescribed

2. The relevant checkboxes as indicated in bold should be marked with a cross where applicable.

3. Enter the name and address of each applicant in the spaces provided in paragraph 3.
   Where the applicant is an individual
   - Names of individuals should be indicated in full and the surname or family name should be underlined.
   -The address of each individual should also be furnished in the space provided
   - The checkbox for "For individual applicant" should be marked with a cross.

   Where the applicant is a body corporate
   - Bodies corporate should be designated by their corporate name and country of incorporation and, where appropriate, the state of incorporation within that country should be entered where provided.
   - The address of the body corporate should also be furnished in the space provided.
   - The checkbox for "For corporate applicant" should be marked with a cross.

   Where the applicant is a partnership
   - The details of all partners must be provided. The name of each partner should be indicated in full and the surname or family name should be underlined.
   - The address of each partner should also be furnished in the space provided
   - The checkbox for "For others" should be marked with a cross and the name and address of the partnership should be indicated in the box provided.

4 In the field for "Country", please refer to the standard list of country codes made available by the Registry of Patents and enter the country code corresponding to the country in question.

5. The declaration of priority in paragraph 4 should state the date of the previous filing, the country in which it was made, and indicate the file number, if available. Where the application relied upon in an International Application or a regional patent application e.g. European patent application, one of the countries designated in that application [being one falling under section 17 of the Patents Act] should be identified and the country should be entered in the space provided.

6. Where the applicant or applicants is/are the sole inventor or the joint inventors, paragraph 5 should be completed by marking with a cross the 'YES' checkbox in the declaration (A) and the 'NO' checkbox in the alternative statement (B). Where this is not the case, the 'NO' checkbox in declaration (A) should be marked with a cross and a statement will be required to be filed on Patents Form 8.

7. When an application is made by virtue of section 20(3), 26(6) or 47(4), the appropriate section should be identified in paragraph 6 and the number of the earlier application or any patent granted thereon identified. Applicants proceeding under section 26(6) should identify which provision in rule 27 they are proceeding under. If the applicants are proceeding under rule 27(1)(a), they should also indicate the date on which the earlier application was amended.

8. Where the applicant wishes an earlier disclosure of the invention by him at an International Exhibition to be disregarded in accordance with section 14(4)(c), then the 'YES' checkbox at paragraph 7 should be marked with a cross. Otherwise, the 'NO' checkbox should be marked with a cross.

9. Where in disclosing the invention the application refers to one or more micro-organisms deposited with a depository authority under the Budapest Treaty, then the 'YES' checkbox at paragraph 8 should be marked with a cross. Otherwise, the 'NO' checkbox should be marked with a cross. Attention is also drawn to the Fourth Schedule of the Patents Rules.

10. Where an agent is appointed, the fields for "DETAILS OF AGENT" and "ADDRESS FOR SERVICE IN SINGAPORE" should be completed and they should be the same as those found in the corresponding Patents Form 41. In the event where no agent is appointed, the field for "ADDRESS FOR SERVICE IN SINGAPORE" should be completed, leaving the field for "DETAILS OF AGENT" blank.

11. In the event where an individual is appointed as an agent, the sub-field "Name" under "DETAILS OF AGENT" must be completed by entering the full name of the individual. The sub-field "Firm" may be left blank. In the event where a partnership/body corporate is appointed as an agent, the sub-field "Firm" under "DETAILS OF AGENT" must be completed by entering the name of the partnership/body corporate. The sub-field "Name" may be left blank.

12 Attention is drawn to sections 104 and 105 of the Patents Act, rules 90 and 105 of the Patents Rules, and the Patents (Patent Agents) Rules 2001.

13. Applicants resident in Singapore are reminded that if the Registry of Patents considers that an application contains information the publication of which might be prejudicial to the defence of Singapore or the safety of the public, it may prohibit or restrict its publication or communication. Any person resident in Singapore and wishing to apply for patent protection in other countries must first obtain permission from the Singapore Registry of Patents unless they have already applied for a patent for the same invention in Singapore. In the latter case, no application should be made overseas until at least 2 months after the application has been filed in Singapore, and unless no directions had been issued under section 33 by the Registrar or such directions have been revoked. Attention is drawn to sections 33 and 34 of the Patents Act.

14. If the space provided in the patents form is not enough, the additional information should be entered in the relevant continuation sheet. Please note that the continuation sheets need not be filed with the Registry of Patents if they are not used.

G00002

159159

METHOD AND APPARATUS FOR PERFORMING MODULAR ARITHMETIC

Background of the Invention

Many electronic interactions require the provision of a certain level of security to ensure that the data contained in a message transfer is difficult to intercept and decode, and/or is capable of being verified as being genuine. To achieve these ends, it is possible to encrypt data according to one of many possible schemes. A popular scheme is called public key cryptography (e.g. PGP). Public key cryptography enables a particular message to be encoded according to an individual's private key and a third party's public key – both are long fixed numbers. The message may then be decoded by the third party through use of their private key. In this way, each party may keep their private key secret and thus control who is able to receive and decode any given message.

One of the key elements of encryption systems is the ability to be able to perform modular arithmetic. The basic calculation which is performed may be written as :

$$S = AB \bmod N \tag{1}$$

where A, B and N are large numbers, typically including many hundreds of digits.

Cryptography systems are generally mathematically complex and can pose a high computational overhead on any system which implements them.

Description of the Prior Art

Prior art systems for performing modular arithmetic make use of Montgomery's theorem, which has been used in many software and hardware implementations of modular arithmetic algorithms. Implementations using Montgomery's theorem are able to compute a value for S without first multiplying A and B and then dividing by N. Most of the hardware implementations rely on an iterative approach which decomposes A into k blocks of p bits to limit the size of the hardware operators required. Further advances have used a serial architecture to further reduce the circuit size. Such architectures are generally based around two serial multipliers, FIFO elements and the pre-computation of a constant J_0, such that :

$$J_0.N \equiv -1 \bmod 2^p \tag{2}$$

k and p are both positive integers, and the binary representation of a positive integer X, where $X < 2^{kp}$ may be given by:

$$X = \sum_{i=0}^{k-1} X[i]2^i \tag{3}$$

where 0 $\leq X[i] < 2$, i.e. X may be either 0 or 1.

Throughout this specification, square brackets [ ] refer to a particular bit position in a multi-bit word e.g. X[i] refers to the i^{th} bit of word X. Angle brackets < > refer to a particular block of a multi-bit word e.g. X<i> refers to the i^{th} block of word X. Parentheses ( ) refer to the value of a word at a particular iteration of a loop function e.g. X(i) refers to the value of word X at the i^{th} iteration.

A definition for $X[j:k]$, where $j > k$, is that X is a positive integer having a total length of $j+1-k$ bits, such that $X[j]$ is the MSB and $X[k]$ is the LSB.

The base 2^p representation of X is given by:

$$X = \sum_{i=0}^{k-1} X\langle i\rangle 2^{pi} \tag{4}$$

where 0 $\leq X\langle i\rangle < 2^p$

In the following description, it is assumed that N is an odd integer such that $2^{p(k-1)} < N < 2^{kp}$, and that both A and B are less than N. A p-bit constant, J_0, is thus defined as:

$$J_0.N\langle 0\rangle \equiv -1.\bmod 2^p \tag{5}$$

N is the modulus number which is used in all public key cryptography systems. It is defined as the product of two large prime numbers (i.e. >>2) and must therefore be odd.

The prior art hardware implementation of the Montgomery theorem may be described by the following pseudo-code.

procedure MM-BASIC(A,B,N)
$S(-1) = 0$
for $i = 0$ **to** $k - 1$
    $T = S(i - 1) + A\langle i\rangle B$
    $Y_0 = (T.J_0) \bmod 2^p$
    $S(i) = (T + NY_0)/2^p$
    if $S(i) \geq N$ **then** $S(i) = S(i) - N$
end for

The implementation of this pseudo code in hardware is shown in a simplified form in Figure 1. The architecture is constructed in serial form so that one bit of the solution is generated for each clock cycle. Such an architecture, as opposed to a parallel one, minimises the amount of hardware required at the expense of speed.

The circuit of figure 1 is arranged to receive five different input signals: *A[k]* 200; *B[t]* 205; *S(i-1)* 210; *GE(i-1)* 215; and *N[t]* 220.

Serial Multiplier 110 accepts as inputs, a fixed p-bit word, $A\langle i\rangle$ produced by register 105, and a one-bit data stream *B[t]* 205. It then acts to produce the output, $(A\langle i\rangle.B)$, one bit at a time.

Multiplier 110 is configured internally as shown in Figure 2. The two inputs are the output 340 of register 105 and *B[t]* 205. The two inputs 205, 340 are ANDed together in AND gate 300. The result of this operation is fed into Carry Save Adder 310, along with two other inputs. The first of these other inputs is the carry output (C) derived from the fed back output from p-bit register 315. The other input to the Adder is

derived from the result output (R) of p-bit register 320 which has been divided by 2 in divider 305. Registers 315, 320 are positioned immediately after the Carry Save Adder 310 and each receives one of the twin outputs produced by the adder.

The Carry Save Adder 310 is arranged to transform a sum of three numbers into a sum of two numbers such that :

$$2.C + R = X + Y + Z \tag{6}$$

The Carry Save Adder 310 computes $C(t)$ and $R(t)$ based on the following bitwise Boolean equations.

$$C(t) = (C(t-1) \text{ OR } R(t-1)/2) \text{ AND } (C(t-1) \text{ AND } B[t].A\langle i\rangle) \text{ AND } (R(t-1)/2 \text{ AND } B[t].A\langle i\rangle) \tag{7}$$

$$R(t) = C(t-1) \oplus R(t-1)/2 \oplus B[t].A\langle i\rangle \tag{8}$$

In a simplified notation:

$$C(t),\ R(t) = \text{SERIAL\_MULT}\ (B[t].A\langle i\rangle,\ C(t\text{-}1),\ R(t\text{-}1)) \tag{9}$$

The procedure MM_BASIC, already shown, may be written in a form which shows the serial operations explicitly:

procedure MM-SERIAL(A, B, N)
$S(-1) = 0$
$GE(-1) = 0$
for $i = 0$ **to** k-1
    #computation of Y_0
    for $t = 0$ **to** p-1
        $C_{S1}(t)$, $R_{S1}(t)$ = SERIAL_SUB($C_{S1}(t\text{-}1)$, $GE(i\text{-}1)$. $N[t]$, $S(i\text{-}1)[t]$)
        $C_{M1}(t)$, $R_{M1}(t)$ = SERIAL_MULT($B[t]$. $A\langle i\rangle$, $C_{M1}(t\text{-}1)$, $R_{M1}(t\text{-}1)$)
        $C_{A1}(t)$, $R_{A1}(t)$ = SERIAL_ADD($C_{A1}(t\text{-}1)$, $R_{M1}(t)[0]$, $R_{S1}(t)$)
        $C_{M2}(t)$, $R_{M2}(t)$ = SERIAL_MULT($R_{A1}(t)$. J_0, $C_{M2}(t\text{-}1)$, $R_{M2}(t\text{-}1)$)

$Y_0[t] = R_{M2}(t)$
end for
# mail loop: computation of $S(i)$
for $t = 0$ **to** $kp+p-1$
    $C_{S1}(t), R_{S1}(t)$ = SERIAL_SUB($C_{S1}(t-1), GE(i-1) . N[t], S(i-1)[t]$)
    $C_{M1}(t), R_{M1}(t)$ = SERIAL_MULT($B[t] . A(i), C_{M1}(t-1), R_{M1}(t-1)$)
    $C_{A1}(t), R_{A1}(t)$ = SERIAL_ADD($C_{A1}(t-1), R_{M1}(t)[0], R_{S1}(t)$)
    $C_{M2}(t), R_{M2}(t)$ = SERIAL_MULT($R_{A1}(t) . J_0, C_{M2}(t-1), R_{M2}(t-1)$)
    $C_{A2}(t), R_{A2}(t)$ = SERIAL_ADD($C_{A1}(t-1), R_{M2}(t)[0], R_{A1}(t)$)
    $S(i)[t-p] = R_{A2}(t)$
    $SGE(t)$ = SERIAL_GE($SGE(t-1), N[t-p], S(i)[t-p]$)
end for
$GE(i) = SGE(kp+p-1)$
end for

The total number of clock cycles required to compute the result according to the above scheme is $k(kp+2p)$.

Summary of the Present Invention

In a first broad form the present invention provides Apparatus having inputs A, B and N, and an output S, said apparatus being arranged to perform a modular operation, S=A.B mod N, the apparatus including a 2-stage Carry Save Adder (2-CSA) and a 1-stage Carry Save Adder (1-CSA), the 2-CSA being arranged to receive 5 input signals:

- U_0, being the partial product of N and Y_0;
- U_1, being the subtraction of a previous version of S and U_8 wherein U_8 is either N or 0 depending on the value of the comparison between the result of the previous iteration and N.
- U_2, being the partial product of B with the current version of A;
- U_3, being S/2
- U_4, being the carry output of the 1-CSA;

where result and carry outputs of the 2-CSA form two of three inputs to the 1-CSA, wherein the result (R) output of the 1-CSA is the desired result (S), and the third input to the 1-CSA is a compensation signal arranged to allow S to be calculated without knowing the constant J_0, where $J_0N<0> = -1. \bmod 2^p$, where p is a block length into which A is sub-divided.

In a second broad form, the present invention provides An iterative method of performing a modular operation of S = A.B mod N, where A, B and N are encoded as multi-bit digital words, including the following steps:

a) setting S(-1) to 0, and i to 0
b) setting S(i) to $(S(i-1) + A<i>B + NY_0)/2^p$
c) setting S(i) to (S(i) – N) if S(i) ≥N
d) repeating steps b) and c) k times.

wherein:

i is a loop counter;
k is a number of blocks of p bits length into which A is divided;
$Y_0 = ((T.J_0) \bmod 2^p)$;
$J_0N = -1\bmod2^p$;and

Y_0 is calculated one bit at a time, based on the fact that $(T + NY_0)$ is a multiple of 2^p.

Other features and benefits of the invention will become apparent in the following description of various embodiments of the invention.

Brief Description of the Drawings

For a better understanding of the present invention and to understand how the same may be brought into effect, the invention will now be described by way of example only, with reference to the appended drawings in which:

Figure 1 shows a simplified prior art circuit for implementing modular arithmetic according to Montgomery's theorem;

Figure 2 shows a prior art serial/parallel multiplier or carry save adder;

Figure 3 shows a merged multiplier as used in embodiments of the invention; and

Figure 4 shows a hardware implementation according to an embodiment of the invention.

Detailed Description of the Preferred Embodiments

The present invention retains a serial architecture to accomplish the calculation, but embodiments of the inventions do not require pre-knowledge of the constant, J_0. Embodiments of the invention calculate $Y_0 = ((T \cdot J_0) \bmod 2^p)$ one bit at a time, based on the fact that $(T + NY_0)$ must be a multiple of 2^p. In this way, the complex mathematical functions required to pre-compute J_0 can be dispensed with.

With this implicit knowledge, the procedure MM-BASIC described previously, may now be written as MM-SIMPLE:

1. **procedure** MM-SIMPLE(A, B, N)
2. $S(-1) = 0$
3. **for** $i = 0$ **to** k-1
4. $\quad S(i) = (S(i\text{-}1) + A(i)B + NY_0)/2^p$
5. $\quad$ **if** S(i) $\geq N$ **then** $S(i) = S(i) - N$
6. **end for**

The above serial implementation of MM-SIMPLE is more efficient than the prior art implementation of MM-BASIC as the two multipliers required in the prior art can be merged into a single multiplier in embodiments of the invention. The gain, in terms of fewer components, is a total of $2p$ registers plus the two serial adders 120 and 155. The removal of the need for these components removes a significant amount of circuitry, and thus the resulting architecture requires less space and consumes less power to achieve the same result. It also calculates the result in fewer clock cycles.

Figure 3 shows the resultant hardware implementation which may be used to perform the steps of procedure MM-SIMPLE presented above.

Y_0 is computed bit by bit during the first p cycles of the loop, starting at line 15 of the procedure MM-SERIAL. Assuming that at cycle $q<p$, the bits 0, 1,...q-1 have already been computed, leaving only bit q to be discovered.

According to embodiments of the present invention, if, at cycle q, the LSB of the 2-stage Carry Save Adder shown in Figure 3 is '1', then $N[q:0]$ is added to the intermediate result, and $Y_0[q] = 1$.

This may be proved as follows. At the q^{th} step, the intermediate values from the first Carry Save Adder may be given as :

$$S = 2C + R \qquad (10)$$

$$= (A\langle i\rangle.B[q:0] + Y_0[q-1:0].N[q:0] + S\langle i-1\rangle[q:0])/2^q \qquad (11)$$

Assuming that the q^{th} bit of Y_0 is a '1', then the above equation may be re-written as:

$$S' = (A\langle i\rangle.B[q:0] + (2^q + Y_0[q-1:0]).N[q:0] + S\langle i-1\rangle[q:0])/2^q \qquad (12)$$

$$= S + N[q:0] \qquad (13)$$

As the LSB of N is always 1, since it is a large prime number and, therefore, odd, then from the above equations, it can be seen that the LSBs of S and S' are always inverted. Therefore, it is possible to guarantee that the LSB of the result is 0 in the first p steps by choosing either S or S'. The choice of S' implies that the q^{th} bit of Y_0 must be forced to equal 1.

The above step is repeated at each cycle $q<p$, so that at the end all bits of Y_0 are discovered.

The procedure, MM-SERIAL-SIMPLE shown below is a pseudo-code implementation of an embodiment of the present invention, and is a version of the previously presented MM-SERIAL adapted according to the above results.

```
1.  procedure MM-SERIAL-SIMPLE(A, B, N)
2.  S(-1) = 0
3.  GE(-1) = 0
4.  for i = 0 to k-1
5.          # main loop: computation of S(i)
6.          for t = 0 to kp+p-1
7.                  C_S1(t), R_S1(t) = SERIAL_SUB(C_S1(t-1), GE(i-1) . N[t], S(i-1)[t])
8.                  C_int, R_int = 2-STAGE_CSA(B[t] . A⟨i⟩, C_M(t-1), R_M(t-1)/2, N[t] . Y_0)
9.                  if t < p and R_int[0] = 1 then
10.                         C_M(t), R_M(t) = CSA(N[t:0], C_int, R_int)
11.                         Y_0[t] = 1
12.                 else
13.                         C_M(t), R_M(t) = C_int, R_int
14.                 end if
15.                 S(i)[t-p] = R_M(t)[0]
16.                 SGE(t) = SERIAL_GE(SGE(t-1), N[t-p], S(i)[t-p])
17.         end for
18.         GE(i) = SGE(kp+p-1)
19. end for
```

The conditional statement at line 9 of the above procedure may be considered to trigger a compensation event which, if $t<p$ and $R_{int}[0] = 1$, causes the value of register 525 N_{del} to be applied to the input of the 1-stage CSA (1-CSA) 540. If the condition is not satisfied, then the C and R outputs of the 2-stage CSA (2-CSA) 520 merely feed straight into the 1-CSA and no compensation is performed.

It is the addition of the compensation function which directly removes the need to explicitly compute J_0.

In figure 4, the compensation function is implemented by register 525, AND gate 530, MUX 535. The MUX 535 effectively performs the conditional IF statement of line

9 of MM-SERIAL-SIMPLE, and if $R_{int}[0]$ is equal to 1, then the contents of register 525 is applied to 1-CSA 540.

The above procedure (MM-SERIAL-SIMPLE) is further explained in the procedure below (MM-SERIAL-SIMPLE_enhanced), which includes further details on selected ones of the internal signal nets.

These internal nets are labelled from U_0 to U_8 and directly correspond with selected internal nets shown in Figure 4.

```
procedure MM-SERIAL-SIMPLE_enhanced(A, B, N)
S(-1) = 0
GE(-1) = 0
A_next = A[p-1:0]
for i = 0 to k-1
        # main loop: computation of S(i)
        N_del = 0
        Y_0 = 0
        R = 0
        C = 0
        A_current = A_next
        A_next = A[(i+1)(p-1):(i+1)p]
        for t = 0 to kp+p-1
            U_0 = AND2(N[t], Y_0)
            U_6 = AND1(GE(i-1), N[t])
            U_1 = SUB1(U_6, S(i-1[t])
            U_2 = AND3(B[t], A_current)
            U_3 = R/2
            U_4 = C
            C_int, R_int = 2-STAGE-CSA(U_0, U_1, U_2, U_3, U_4)
            U_7 = MUX(R_int[0],0)
            U_5 = AND4(U_7, N_del)
            if t < p then
                Y_0[t] = U_7
                N_del[t] = N[t]
```

```
            U8 = 0
            else
            # Ndel acts as a shift register
            U8 = Ndel[0]
            Ndel = Ndel/2
            Ndel[p-1] = N[t]
        endif
        C,R = CSA(U5, Cint, Rint)
        S(i)[t] = R[0]
        SGE(t) = GE(U8, R[0])
    end for
    GE(i) = SGE(kp+p-1)
end for
```

As an example, presented below are details of how an embodiment of the invention operates on some sample input data. The following inputs are provided, in 32-bit format:

A = C7197F0E

B = CCEFBAE4_77AF9EE5_848D8AE6

N = D077EC53_F4AA27A4_D7816723

The result of the Montgomery multiplication of *A* by *B* is given by $(AB+NY_0)/2^p$. Before the computation starts, the registers of the multiplier are initialised as follows.

N_0 =	*00000003*
Y_0 =	*00000000*
RC =	*0_00000000*
B[t] =	*6*
N[t] =	*3*

For the sake of simplicity, the registers *R* and *C* have been summed into register RC, and the computation is performed 4 bits (a nibble) at the time, thus setting *p*=4.

1. Computation of the intermediate results, based on the partial products

N[t].Y0 = *0_00000000*
+B[t].A = *4_AA98FA54*
+RC/16 = *0_00000000*
=Intermediate = *4_AA98FA54*

2. Find the first 4 bits of compensation value (*Z*) such that the 4 LSBs of *Intermediate*+$Z.N_0$ are all zero.

   Z = *4*

3. Add the partial product $Z.N_0$ to *Intermediate*

   Intermediate = *4_AA98FA54*
   $+Z.N_0$ = *0_0000000C*
   =RC = *4_AA98FA60*

4. Update the registers with the new values and restart the cycle

N_0 = *00000023* Y_0 = *00000004* *RC* = *4_AA98FA60* *B[t]* = *E* *N[t]* = *2*

1. Computation of the intermediate results, based on the partial products

   N[t].Y0 = *0_0000008C*
   +B[t].A = *A_E364F2C4*
   +RC/16 = *0_4AA98FA6*
   =Intermediate = *B_2E0E8272*

2. Find first 4 bits of compensation (*Z*) such that the 4 lsb of *Intermediate*+$Z.N_0$ are all zero.

   Z = *A*

3. Add the partial product $Z.N_0$ to *Intermediate*

   Intermediate = *B_2E0E8272*
   $+Z.N_0$ = *0_0000015E*
   =RC = *B_2E0E83D0*

4. Update the registers with the new values and restart the cycle

N_0 = *00000723* Y_0 = *000000A4* *RC* = *B_2E0E83D0* *B[t]* = *A* *N[t]* = *7*

1. Computation of the intermediate results, based on the partial products

$N[t].Y0$	$= 0\_0000047C$
$+B[t].A$	$= 7\_C6FEF68C$
$+RC/16$	$= 0\_B2E0E83D$
$=Intermediate$	$= 8\_79DFE345$

2. Find first 4 bits of compensation (Z) such that the 4 lsb of *Intermediate*+Z.N0 are all zero.

$Z \quad = 9$

3. Add the partial product $Z.N_0$ to *Intermediate*

Intermediate	$= 8\_79DFE345$
$+Z.N_0$	$= 0\_0000403B$
$=SUM_2$	$= 8\_79E02380$

4. Update the registers with the new values and restart the cycle

$N_0 = 00006723$ $Y_0 = 000009A4$ $RC = 8\_79E02380$ $\quad B[t] = 8 \quad N[t] = 6$

This process is repeated until all the bits of Y_0 are discovered. At this stage, the compensation phase is no longer needed so the computation iterates over the remaining bits of *B* and *N*. The step by step result at each phase is given by the following table:

Cycle	N_0	Y_0	*RC*	*B[t]*	*N[t]*
0	XXXXXXXX	XXXXXXXX	XXXXXXXXXX	X	X
1	00000003	00000000	0000000000	6	3
2	00000023	00000004	04AA98FA60	E	2
3	00000723	000000A4	0B2E0E83D0	A	7
4	00006723	000009A4	0879E02380	8	6
5	00016723	000009A4	06C06A3480	D	1
6	00816723	000A09A4	0A88602800	8	8
7	07816723	000A09A4	06E1A24810	4	7
8	D7816723	090A09A4	03CE530470	8	D
9	D7816723	790A09A4	0CCFBD7800	5	4
10	4D781672	790A09A4	0694A37956	E	A
11	A4D78167	790A09A4	1007138AC1	E	7
12	7A4D7816	790A09A4	0F331C6EEC	9	2

13	27A4D781	790A09A4	08E52B51B4	F	A
14	A27A4D78	790A09A4	10F3358755	A	A
15	AA27A4D7	790A09A4	0D9096AF69	7	4
16	4AA27A4D	790A09A4	082EE40AE8	7	F
17	F4AA27A4	790A09A4	0D0C374AAC	4	3
18	3F4AA27A	790A09A4	0558478DCE	E	5
19	53F4AA27	790A09A4	0D961B9BD4	A	C
20	C53F4AA2	790A09A4	0E4CD923F9	B	E
21	EC53F4AA	790A09A4	1011728ED1	F	7
22	7EC53F4A	790A09A4	0FFADBDE3B	E	7
23	77EC53F4	790A09A4	0F3258F423	C	0
24	077EC53F	790A09A4	0A485783EA	C	D
25	D077EC53	790A09A4	101F39EA3A	0	0
26	0D077EC5	790A09A4	0101F39EA3	0	0
27	00D077EC	790A09A4	00101F39EA	0	0
28	000D077E	790A09A4	000101F39E	0	0
29	0000D077	790A09A4	0000101F39	0	0
30	00000D07	790A09A4	00000101F3	0	0
31	000000D0	790A09A4	000000101F	0	0
32	0000000D	790A09A4	0000000101	0	0
33	00000000	790A09A4	0000000010	0	0
34	00000000	790A09A4	0000000001	0	0

Notice that the serial output result can be read directly as the right most nibble of the RC column. It is also interesting to notice the shifting pattern of N_0. From cycle 1 to 8, the register behavior is comparable to a stack, where the nibble are pushed from the left. From cycle 9 onward, the register behaves as a right shift register. The output of this register shall be used as the input of a comparator which detects if the results is greater or equal to N.

Y = 790A09A4

RESULT = 1_01F39EA3_AA3B194E_C8954C16_00000000

In the light of the foregoing description, it will be clear to the skilled man that various modifications may be mode within the scope of the invention.

The present invention includes and novel feature or combination of features disclosed herein either explicitly or any generalisation thereof irrespective of whether or not it relates to the claimed invention or mitigates any or all of the problems addressed.

CLAIMS

1. Apparatus having inputs A, B and N, and an output S, said apparatus being arranged to perform a modular operation, S=A.B mod N, the apparatus including a 2-stage Carry Save Adder (2-CSA) and a 1-stage Carry Save Adder (1-CSA), the 2-CSA being arranged to receive 5 input signals:

- U_0, being the partial product of N and Y_0;
- U_1, being the subtraction of a previous version of S and U_6 wherein U_6 is either N or 0 depending on the value of the comparison between the result of the previous iteration and N.
- U_2, being the partial product of B with the current version of A;
- U_3, being S/2
- U_4, being the carry output of the 1-CSA;

where result and carry outputs of the 2-CSA form two of three inputs to the 1-CSA, wherein the result (R) output of the 1-CSA is the desired result (S), and the third input to the 1-CSA is a compensation signal arranged to allow S to be calculated without knowing the constant J_0, where $J_0N<0> = -1. \bmod 2^p$, where p is a block length into which A is sub-divided.

2. Apparatus as claimed in claim 1 wherein the compensation signal is arranged to equal a delayed version of N in the event that t<p and the Result (R) output of the 2-CSA equals '1'.

3. Apparatus as claimed in any one of the preceding claims wherein the 2-CSA includes two 1-CSA arranged in series.

4. Apparatus as claimed in any one of the preceding claims wherein while processing bits 0 to p-1, register Y_0 is arranged such that the LSB of the Result (R) output of the 1-CSA is always '0'.

5. Apparatus as claimed in any one of the preceding claims wherein the apparatus is arranged to take the form of a custom integrated circuit.

6. Apparatus as claimed in claim 5 wherein the custom integrated circuit includes a digital signal processor (DSP).

7. An iterative method of performing a modular operation of S = A.B mod N, where A, B and N are encoded as multi-bit digital words, including the following steps:

a) setting S(-1) to 0, and i to 0

b) setting S(i) to $(S(i\text{-}1) + A\langle i\rangle B + NY_0)/2^p$

c) setting S(i) to (S(i) – N) if S(i) $\geq$N

d) repeating steps b) and c) k times.

wherein:

i is a loop counter;

k is a number of blocks of p bits length into which A is divided;

$Y_0 = ((T.J_0) \bmod 2^p)$;

$J_0N = -1 \bmod 2^p$;and

Y_0 is calculated one bit at a time, based on the fact that $(T + NY_0)$ is a multiple of 2^p.

ABSTRACT

METHOD AND APPARATUS FOR PERFORMING MODULAR ARITHMETIC

An apparatus and method is disclosed for performing the modular operation S=AB mod N. The apparatus is arranged such that the constant J0 which is ordinarily required in order to complete the operation is not required to be explicitly computed, thus simplifying and speeding up the operation.

Figure 4

G00002

G00002

163163

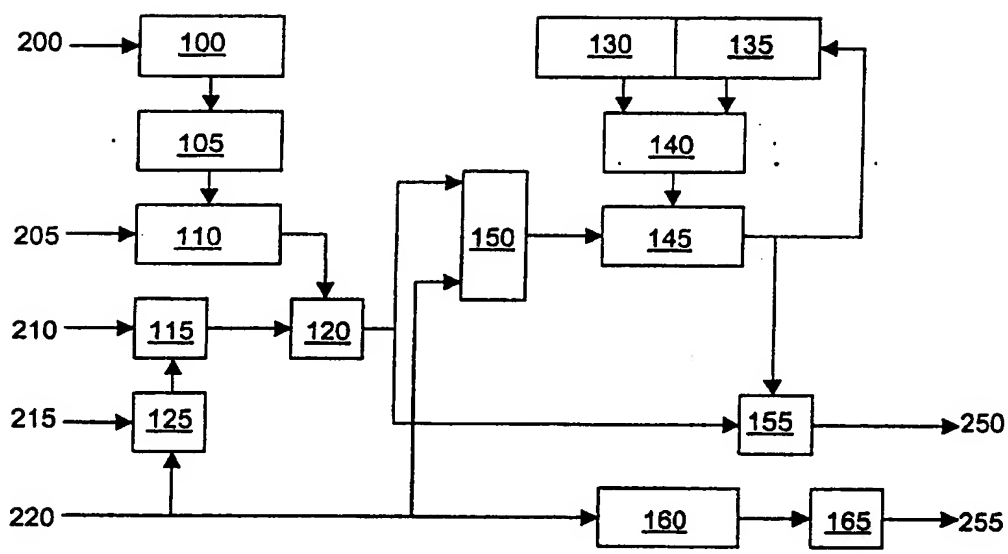


FIGURE 1

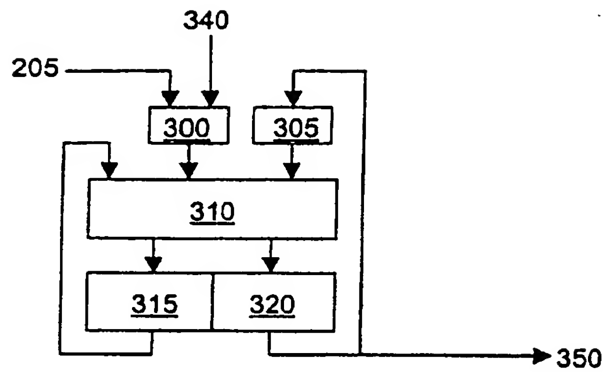


FIGURE 2

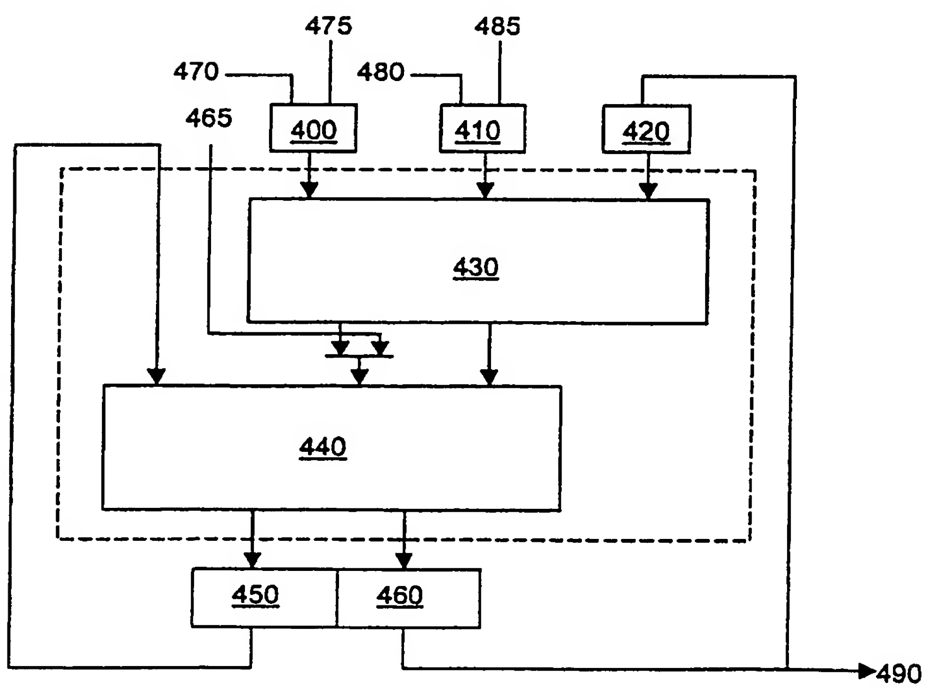
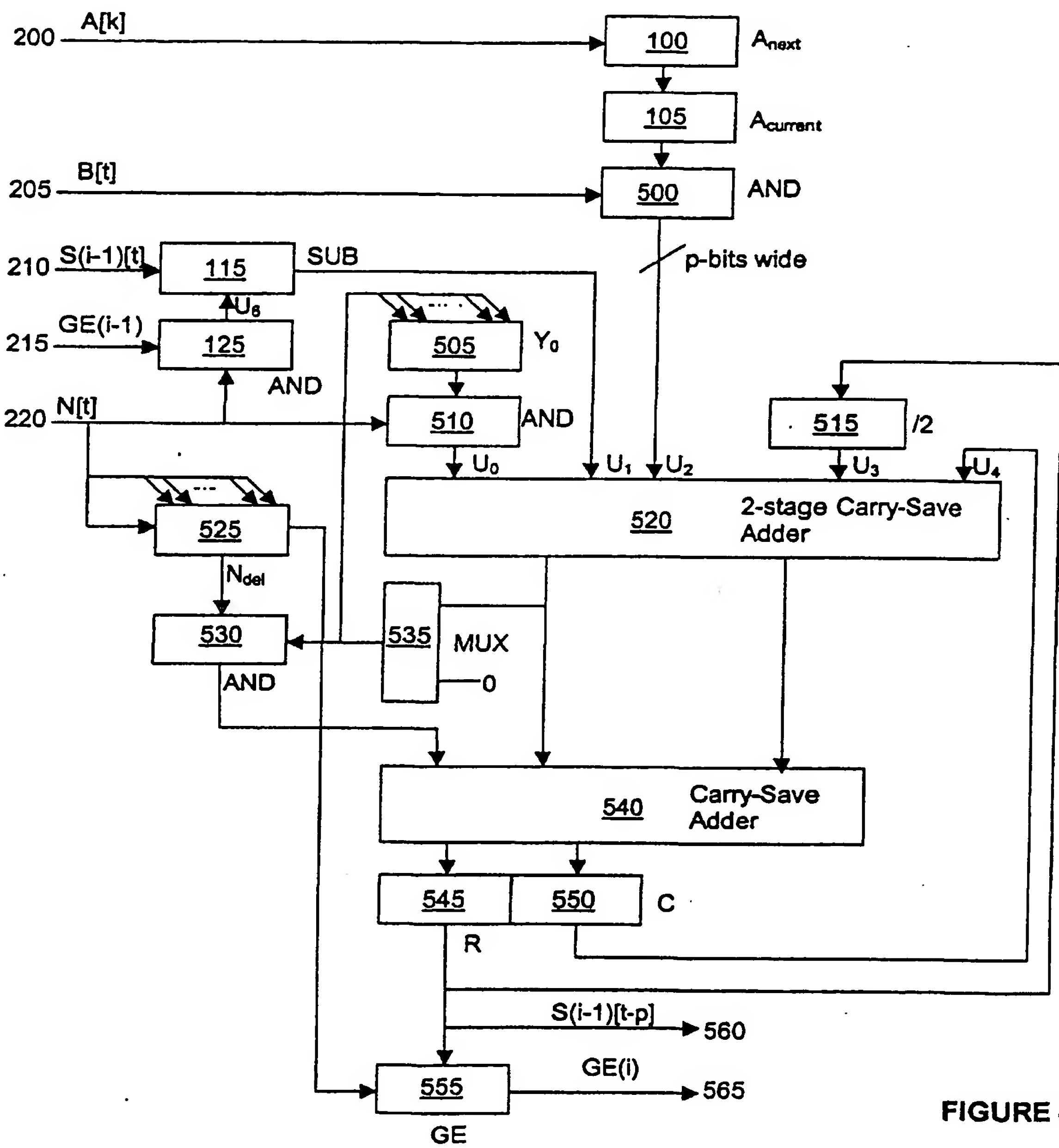


FIGURE 3



200
A[k]
100
A_{next}
105
$A_{current}$
205
B[t]
500
AND
p-bits wide
210
S(i-1)[t]
115
SUB
U_8
215
GE(i-1)
125
AND
505
Y_0
220
N[t]
510
AND
515
/2
U_0
U_1
U_2
U_3
U_4
520
2-stage Carry-Save Adder
525
N_{del}
530
AND
535
MUX
0
540
Carry-Save Adder
545
550
C
R
S(i-1)[t-p]
560
GE(i)
565
555
GE

FIGURE 4

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☑ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: ______________________

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.